Causal Machine Learning (III): Causal Forest

Prof. Tzu-Ting Yang
楊子霆

Institute of Economics, Academia Sinica
中央研究院經濟研究所

May 14, 2026

Main Idea

Treatment Effect Heterogeneity

- In previous lectures, we focused on one average causal parameter — the **Average Treatment Effect (ATE)**:

$$\alpha_{\text{ATE}} = \mathrm{E}\Big[Y_i^1 - Y_i^0\Big]$$

- But treatment effects may **vary across individuals**: a job training program may help young low-education workers more than older workers

- Causal forest estimates the **Conditional Average Treatment Effect (CATE)**:

$$\tau(x) = \mathrm{E}\Big[Y_i^1 - Y_i^0 \mid X_i = x\Big]$$

- It answers the policy question: **who benefits more** from a program?

Motivating Example: Job Training

- Suppose a government offers a job training program.
- Outcome Y_i: earnings after the program.
- Treatment D_i: whether worker i received training.
- Covariates X_i: age, education, race, marital status, previous earnings.

Main policy question

Does the training program help everyone equally, or does it help some workers more than others?

Why Heterogeneity Matters

Worker type	Control earnings	Training earnings	Effect
Young, low education	\$8,000	\$11,000	\$3,000
Older, high experience	\$18,000	\$18,500	\$500
Young, low experience	\$2,000	\$6,000	\$4,000

- The average effect may look positive but modest.
- Policy decisions often require knowing **where** the effect is large.
- CATE helps answer: who should be prioritized if resources are limited?

Why Heterogeneity Matters

- Youth employment programs: which youth benefit most?
- Education policy: do returns differ by background or local labor market?

- Minimum wage: do effects differ by worker skill, firm size, or region?
- Parental leave: do effects differ by previous earnings or occupation?

Treatment Effect Heterogeneity

Traditional Approaches

- Researchers have traditionally explored heterogeneity in two ways:
  1. **Theory-based subgroup analysis**: guided by economic theory, split the sample by gender, age group, or education level and estimate the ATE in each subgroup
  2. **Interaction terms in regression**:

$$Y_i = \alpha + \beta D_i + \gamma X_i + \delta (D_i \times X_i) + \varepsilon_i$$

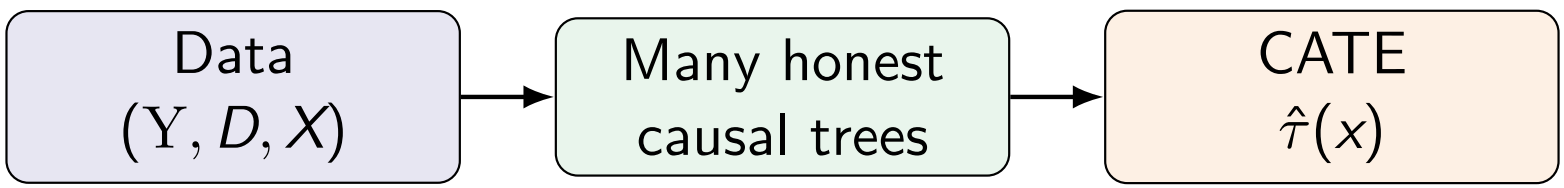
     the coefficient δ captures how the effect of D_i varies linearly with X_i

- Both require the researcher to **pre-specify** which X matters and assume a **linear, parametric** form for heterogeneity

Treatment Effect Heterogeneity

Causal Forest: A Data-Driven Approach

- Instead of estimating one average effect, we want a function $\tau(x)$ that varies with observed characteristics X_i
- A simple regression of Y_i on D_i and X_i gives a linear, parametric answer — it cannot capture flexible heterogeneity
- Causal forest adapts the idea of **random forest** to the causal inference setting:



- Each tree uses a **random subset** of observations and covariates; averaging trees removes noise and gives a stable $\hat{\tau}(x)$

- Causal forest is **not** an identification strategy — it does not solve the selection bias problem on its own
- It is a flexible, **data-driven estimator** of CATE that can be paired with any identification strategy:
  - **RCT**: who benefits more from a randomized treatment?
  - **DID**: are treatment effects heterogeneous across groups?
  - **RDD**: who benefits most near the discontinuity threshold?
- Unlike subgroup analysis or interaction terms, causal forest **searches the data** for which characteristics drive heterogeneity
  - Without the researcher pre-specifying them

How ATE and ATT Relate to CATE

- Once we know the CATE function $\tau(x)$, familiar estimands are just averages of it:

$$\alpha_{\text{ATE}} = \mathrm{E}[\tau(X_i)]\,, \qquad \alpha_{\text{ATT}} = \mathrm{E}[\tau(X_i) \mid D_i = 1]\,.$$

- Causal forest therefore does not *replace* ATE or ATT.
  - It estimates a richer object $\tau(x)$ first, and policy summaries such as ATE or ATT are recovered by averaging.
- This also means: if heterogeneity is small, $\tau(x) \approx \alpha_{\text{ATE}}$ for all x

Decision Tree and Random Forest

Decision Trees

Decision Trees

Main Idea

- A **decision tree** partitions the covariate space into rectangular regions, and fits a simple model (e.g., a mean) within each region
- **Algorithm :**
  1. Find the variable X^j and split point s that minimizes prediction error
  2. Divide the data into two regions: $\{X^j < s\}$ and $\{X^j \geq s\}$
  3. Repeat within each region until a stopping rule is met
- The final estimate in each **leaf** (terminal node) is the mean of Y for observations that fall in that leaf

How to find (X^j, s) that minimises prediction error:

1. For each variable X^j and each candidate split point s (every observed value):
   - Divide data into two regions:
     $R_1(j,s) = \{i : X_i^j < s\}$ and $R_2(j,s) = \{i : X_i^j \geq s\}$
   - Compute the **residual sum of squares (RSS)** after the split:

$$\text{RSS}(j,s) = \sum_{i \in R_1} \left(Y_i - \bar{Y}_{R_1}\right)^2 + \sum_{i \in R_2} \left(Y_i - \bar{Y}_{R_2}\right)^2$$

     where $\bar{Y}_{R_1}, \bar{Y}_{R_2}$ are the mean Y in each region
2. Choose the (j^*, s^*) that gives the **smallest** RSS(j,s)
3. Repeat recursively inside each region

Stopping rules control tree complexity and prevent overfitting:

- **Minimum leaf size**:
  - Do not split a region if it contains fewer than k observations (e.g. $k = 5$).
  - Ensures each leaf has enough data for a reliable estimate of $\bar{Y}$.
- **Maximum tree depth**:
  - Halt after d levels of recursive splits (e.g. $d = 5$).
  - Limits the total number of leaves to at most 2^d.

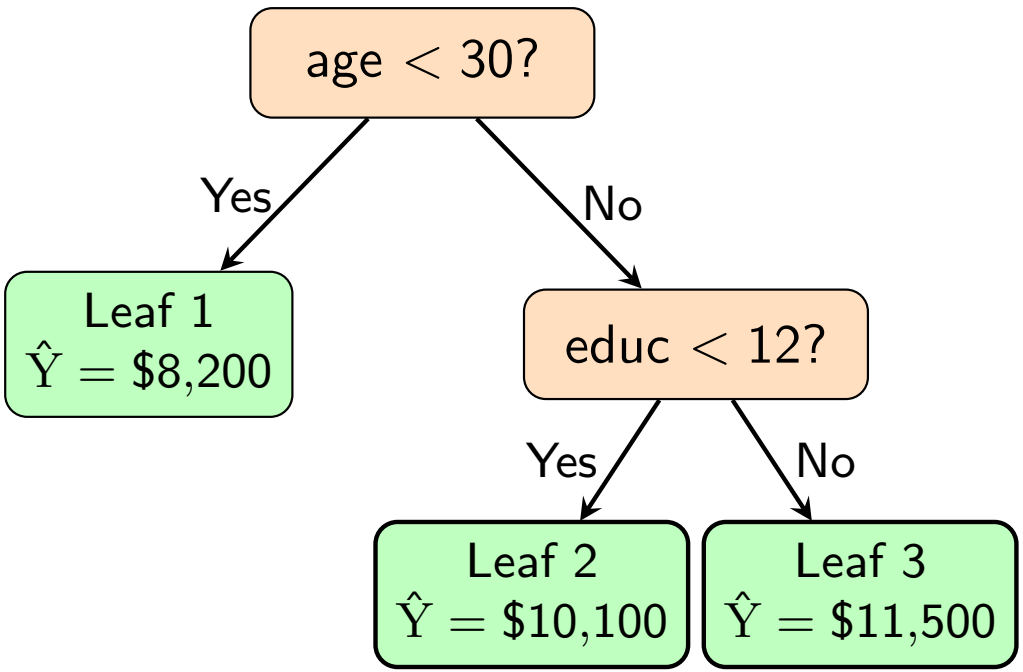
- **Minimum RSS reduction**:
  - Stop splitting a region if the best available split reduces RSS by less than ε.
  - Skips splits that add complexity without meaningfully improving fit.
- **Why necessary?**
  - A tree that splits until every leaf has exactly one observation fits the training data *perfectly* but has no predictive power — this is **overfitting**

Decision Trees

Illustration

- Suppose we want to predict wages Y using age and education X
  - Split 1: age < 30 vs. age ≥ 30
  - Split 2 (within age ≥ 30): educ < 12 vs. educ ≥ 12
- Each leaf contains observations with similar X and similar predicted Y

Decision Trees: Tree Structure Example

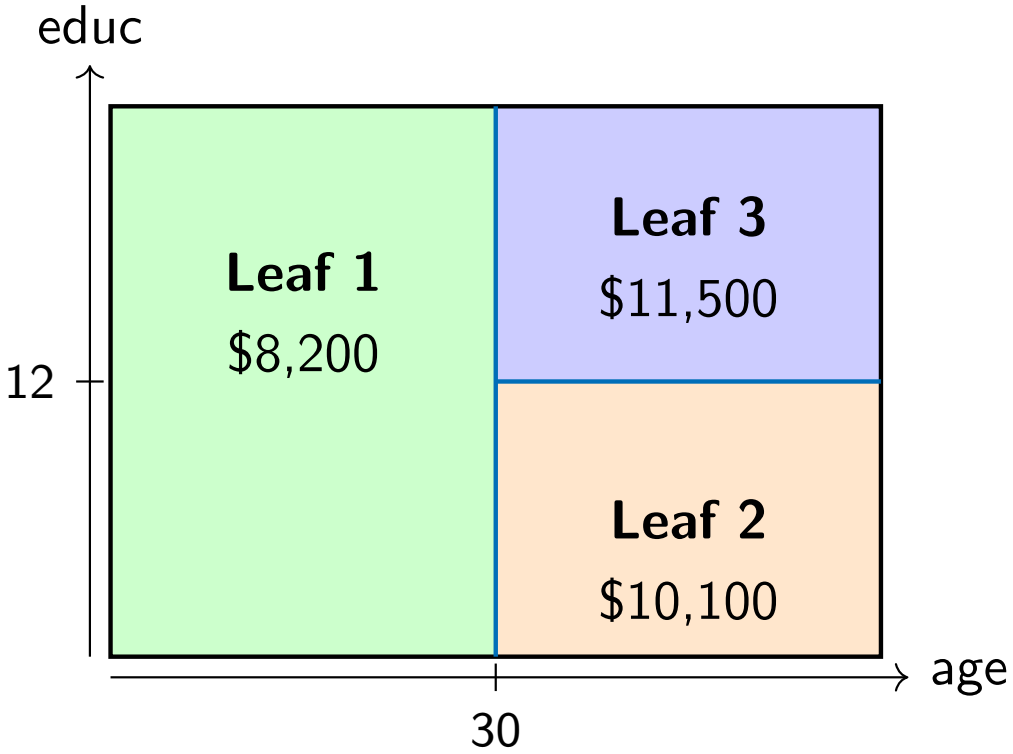


- **Internal nodes** (orange): decision rules on X
- **Leaves** (green): prediction $\hat{Y} =$ mean of Y_i in that region
- **Prediction**: follow branches until reaching a leaf
- *Example*: age $= 35$, educ $= 14$

  $\rightarrow$ age ≥ 30, educ ≥ 12

  $\rightarrow$ **Leaf 3**: $\hat{Y} = \$11{,}500$

Decision Trees: Partition of Feature Space



- Each split divides the feature space into **rectangular regions**
- All observations in the same region receive the **same prediction** (the in-region mean of Y)
- A **deeper tree** $\Rightarrow$ more splits $\Rightarrow$ smaller regions $\Rightarrow$ more flexible, but risks **overfitting**
- The tree approximates any non-linear relationship between Y and X without specifying a functional form

Random Forest

Why a Single Decision Tree Is Not Enough

- A single deep tree has a fundamental weakness: **high variance**
  - The tree "memorizes" the training data rather than learning the true pattern
  - Small changes in the data (e.g. adding or removing a few observations) can lead to a completely different tree structure
  - Predictions on new data are unstable and often poor

Why a Single Decision Tree Is Not Enough

High Variance

Sample A

Age	Educ	Wage
20	9	\$8
25	9	\$9
40	9	\$21
45	9	\$22
50	12	\$23

- Age < 30: wages \$8–9
- Age ≥ 30: wages \$21–23

$\Rightarrow$ **Tree splits on Age**

Sample B

Age	Educ	Wage
20	9	\$8
25	12	\$30
40	9	\$21
45	9	\$22
50	12	\$23

- Age < 30: wages \$8 and \$30
- Age ≥ 30: wages \$21–23

$\Rightarrow$ **Tree switches to Education**

Why a Single Decision Tree Is Not Enough

High Variance

Why does the tree change its split?

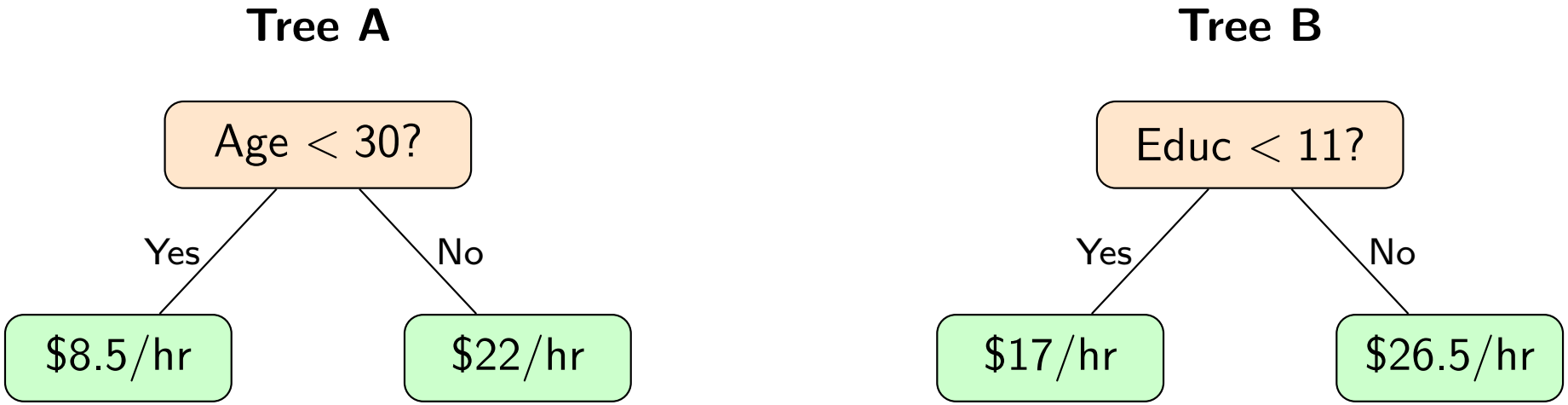
Candidate split	RSS in Sample A	RSS in Sample B
Age < 30	**2.5**	244
Educ < 11	170	**147**

- The tree always picks the split with the lowest RSS:
  - Replacing 1 worker causes the RSS of the age split to explode: $2.5 \rightarrow 244$

$\Rightarrow$ **High variance**: a tiny data change leads to a completely different tree

Why a Single Decision Tree Is Not Enough

High Variance



- Tree A splits on **Age**; Tree B splits on **Education**
- Changing only one worker makes different trees

Why a Single Decision Tree Is Not Enough

High Variance

- Suppose we want to predict new worker's wage: Age $= 22$, Education $= 16$ years

	Decision path	Predicted wage
Tree A	Age $< 30 \rightarrow$ **Yes**	\$8.5/hr
Tree B	Educ $< 11 \rightarrow$ **No**	\$30/hr

- Same worker, same features: yet predictions differ by \$21.5/hr
  - This is **high variance**: replacing just 1 training observation leads to wildly different predictions
  - Solution: average over many trees grown on different subsamples $\Rightarrow$ **Random Forest**

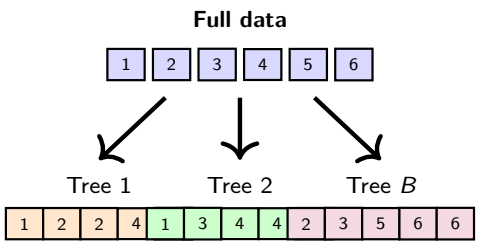
Random Forest: How It Works

Two Sources of Randomness

Random Forest makes trees different from one another in two ways.

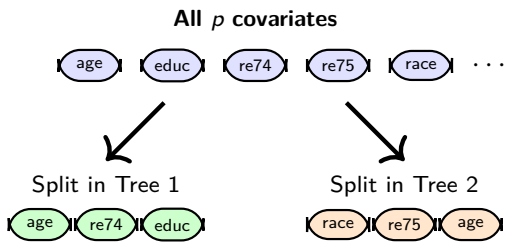
1. Random subsampling

- Draw a random subset of observations for each tree
- Each tree sees a different version of the data
- Some observations are used often; some are left out



2. Random feature selection

- At each split, consider only m randomly selected covariates
- Typical choice: $m = \sqrt{p}$
- Prevents all trees from using the same strongest variable



Result: trees are useful but not identical ⇒ their errors partly cancel when averaged.

Random Forest: How It Works

Algorithm

For a new observation with covariates x:

1. Grow B different decision trees.
   - Each tree uses a random subsample of observations.
   - Each split considers a random subset of covariates.
2. Let each tree produce one prediction:

$$\hat{Y}_1(x),\ \hat{Y}_2(x),\ \ldots,\ \hat{Y}_B(x).$$

3. Average all tree predictions:

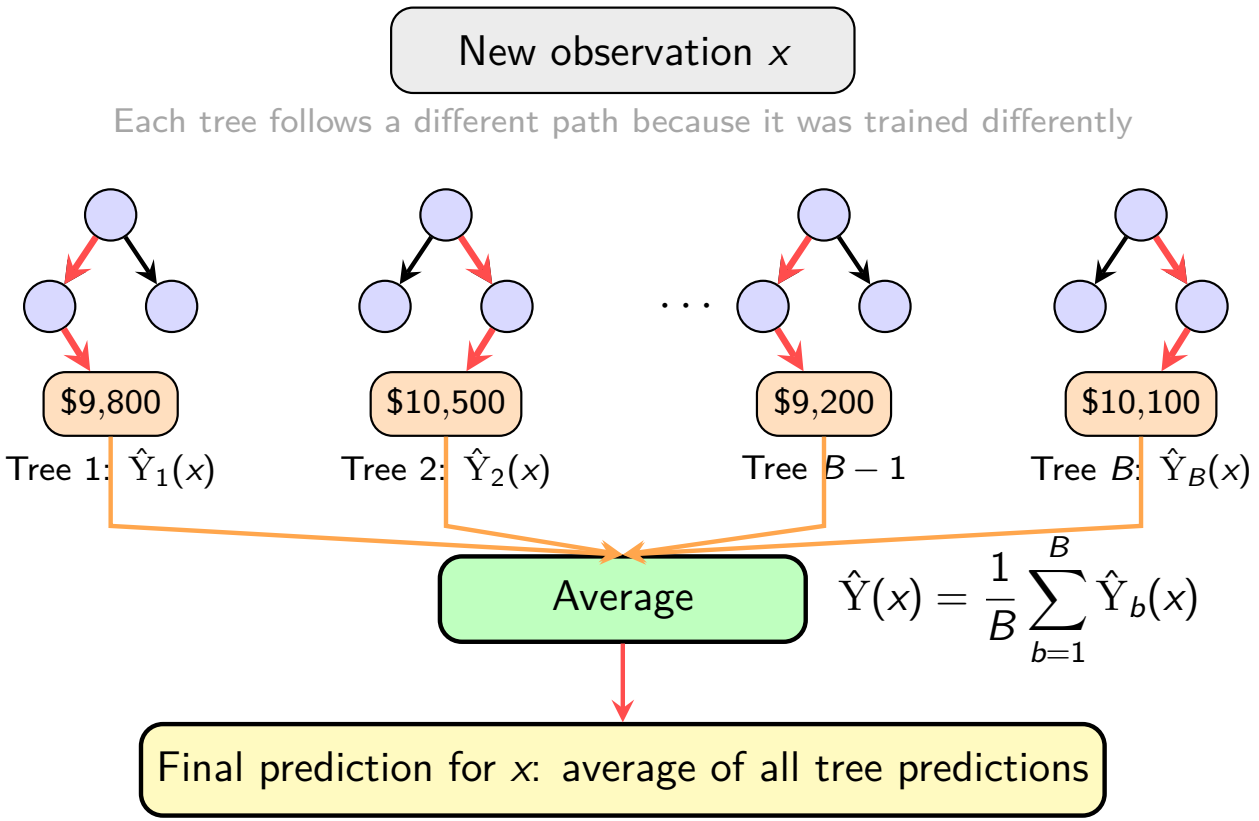
$$\hat{Y}(x) = \frac{1}{B}\sum_{b=1}^{B} \hat{Y}_b(x).$$

Key intuition

A single tree is unstable; many diverse trees averaged together are much more stable.

Random Forest: Illustration

Many Trees, One Averaged Prediction



Causal Tree

From Prediction Tree to Causal Tree

Same Tree Logic, Different Target

Prediction tree

- Goal: predict outcome Y_i
- Split rule: find groups with different outcome levels
- Leaf value:

$$\hat{\mathrm{Y}}(\ell) = \bar{Y}_\ell$$

Causal tree

- Goal: estimate treatment effect $\tau(x)$
- Split rule: find groups with different treatment effects
- Leaf value:

$$\hat{\tau}(\ell) = \bar{Y}^1_\ell - \bar{Y}^0_\ell$$

Key analogy: prediction trees search for outcome heterogeneity; causal trees search for treatment-effect heterogeneity.

In each leaf, compare treated and control observations.

Leaf ℓ	Treated mean	Control mean	Leaf effect
Young, low education	$\bar{Y}_\ell^1 = \$11{,}000$	$\bar{Y}_\ell^0 = \$8{,}000$	$\hat{\tau}(\ell) = \$3{,}000$
Older, high experience	$\bar{Y}_\ell^1 = \$18{,}500$	$\bar{Y}_\ell^0 = \$18{,}000$	$\hat{\tau}(\ell) = \$500$

- A new worker with covariates x falls into one leaf $\ell(x)$.
- The tree predicts that worker's CATE using the leaf comparison:

$$\hat{\tau}(x) = \hat{\tau}(\ell(x)).$$

- Interpretation: workers with similar characteristics have similar treatment effects.

Causal Tree

Finding the Best Split

Prediction tree: choose splits that make outcomes within each leaf more similar

Causal tree: choose splits that make treatment effects across leaves more different

1. For a candidate split (X^j, s), define two regions:

$$R_1 = \{i : X_i^j < s\}, \qquad R_2 = \{i : X_i^j \geq s\}.$$

2. Estimate treatment effects in each region:

$$\hat{\tau}(R_k) = \bar{Y}^1_{R_k} - \bar{Y}^0_{R_k}, \qquad k = 1, 2.$$

3. Prefer splits with large differences:

$$\Delta(j, s) = \left|\hat{\tau}(R_1) - \hat{\tau}(R_2)\right|.$$

Causal Tree

A Small Split Example

Question: which split best separates high-effect and low-effect workers?

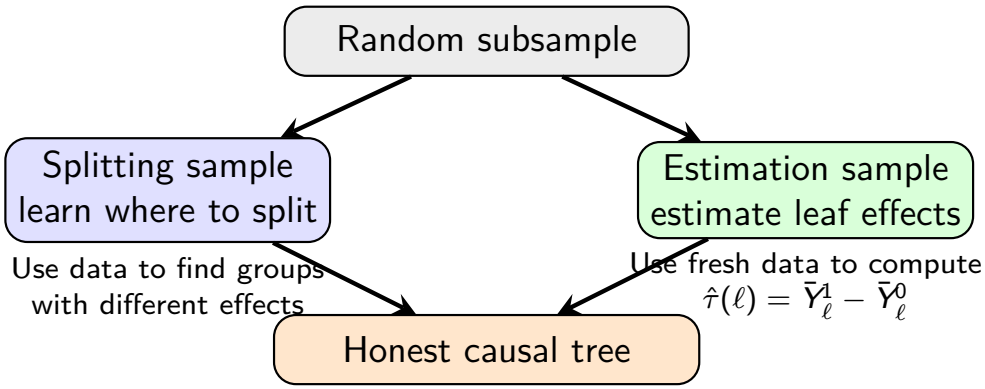
Split	$\hat{\tau}(R_1)$	$\hat{\tau}(R_2)$	Δ
Age < 30	\$3,000	\$600	**\$2,400**
Education < 12	\$2,100	\$1,400	\$700
Prior earn. $< 5{,}000$	\$2,600	\$1,000	\$1,600

- In this toy example, the tree first splits on **Age** < 30.
- Not because young and old workers have the most different earnings levels.
- But because their **treatment effects** are most different.

Causal Tree

Honest Estimation

In practice, causal trees use separate data for two different jobs.



- The splitting sample answers: **which subgroups should we compare?**
- The estimation sample answers: **how large is the treatment effect in each subgroup?**
- This avoids using the same observations to both discover a pattern and report its size.

Why "honest"? The effect is estimated using data that did not choose the split.

Causal Forest

From Random Forest to Causal Forest

Same Averaging Logic

Random Forest

- Build many prediction trees
- Each tree predicts outcome:

$$\hat{Y}_b(x)$$

- Average predictions:

$$\hat{Y}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{Y}_b(x)$$

Causal Forest

- Build many honest causal trees
- Each tree estimates CATE:

$$\hat{\tau}_b(x)$$

- Average treatment effects:

$$\hat{\tau}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{\tau}_b(x)$$

Random forest stabilizes outcome prediction; causal forest stabilizes CATE estimation.

Causal Forest: How It Works

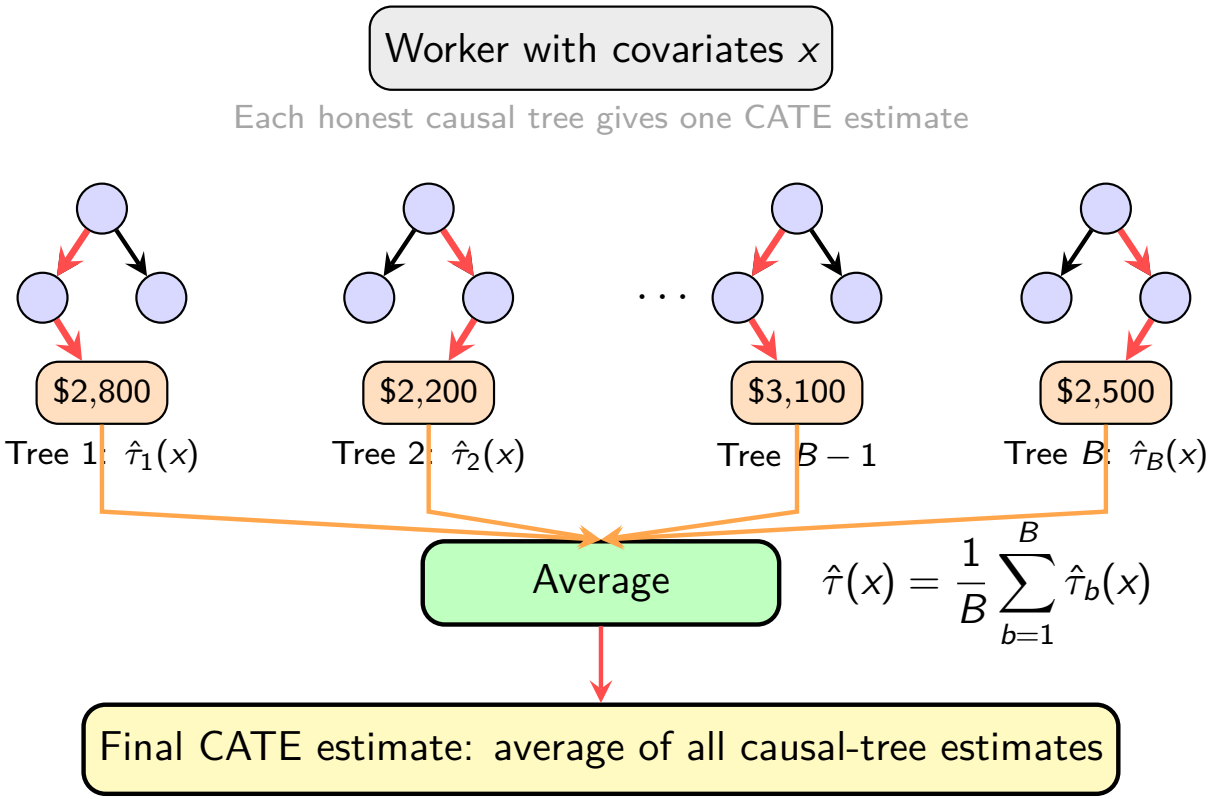
Algorithm

1. Draw a random subsample of observations.
2. Split that subsample into two parts:
   - **splitting sample**: choose splits that maximize treatment-effect heterogeneity;
   - **estimation sample**: estimate treatment effects in leaves.
3. Grow one honest causal tree.
4. Repeat for B trees.
5. For a worker with characteristics x, average all tree-specific CATE estimates:

$$\hat{\tau}(x) = \frac{1}{B} \sum_{b=1}^{B} \hat{\tau}_b(x).$$

Causal Forest: Illustration

Many Causal Trees, One CATE Estimate



Why Does Causal Forest Help?

Same Benefit as Random Forest

- A single causal tree is easy to explain, but it is noisy.
- Different subsamples may produce different splits and different leaf effects.
- Causal forest averages many honest causal trees:

$$\hat{\tau}(x) = \tau(x) + \frac{1}{B}\sum_{b=1}^{B} \varepsilon_b(x).$$

- If tree-specific errors partly cancel, the average is more stable.

Random forest: average noisy outcome predictions.
Causal forest: average noisy treatment-effect estimates.

Causal Forest

What Do We Get After Estimation?

- **Individual CATE estimates**: $\hat{\tau}(x_i)$ for each worker.
- **ATE**: average the CATE over all observations:

$$\hat{\alpha}_{\text{ATE}} \approx \frac{1}{N} \sum_{i=1}^{N} \hat{\tau}(x_i).$$

- **ATT**: average the CATE among treated workers:

$$\hat{\alpha}_{\text{ATT}} \approx \frac{1}{N_1} \sum_{i:D_i=1} \hat{\tau}(x_i).$$

- **Heterogeneity summaries**: CATE distribution, BLP, variable importance.

Causal Forest

How to Interpret $\hat{\tau}(x_i)$

- $\hat{\tau}(x_i)$ is **not** the true individual treatment effect.
- We still observe only one potential outcome for worker i.
- Better interpretation:
  - $\hat{\tau}(x_i)$ is the estimated average treatment effect for workers with characteristics similar to x_i.

Example

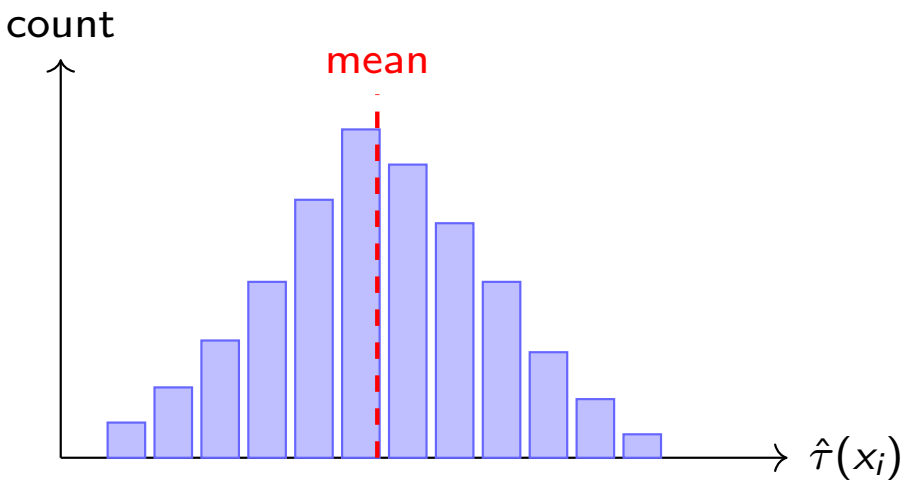
If $\hat{\tau}(x_i) = \$2{,}500$, workers similar to worker i are estimated to gain about \$2,500 from job training.

Causal forest estimates CATE, not each person's true counterfactual outcome.

After Causal Forest

CATE Distribution

Question: How much do estimated treatment effects vary across workers?



- Plot all individual CATE estimates $\hat{\tau}(x_i)$.
- The center of the distribution is related to the ATE.
- A **wide** distribution means treatment effects differ substantially across workers.
- A **narrow** distribution means most workers have similar estimated effects.

Use: tells us whether policy targeting may be valuable.

After Causal Forest

Best Linear Projection (BLP)

Question: Which worker characteristics are associated with larger estimated treatment effects?

- CATE estimates can be hard to summarize one by one.
- BLP gives a simple linear summary:

$$\hat{\tau}(x_i) = \beta_0 + \beta_1 \,\text{age}_i + \beta_2 \,\text{educ}_i + \beta_3 \,\text{re74}_i + \cdots + \varepsilon_i.$$

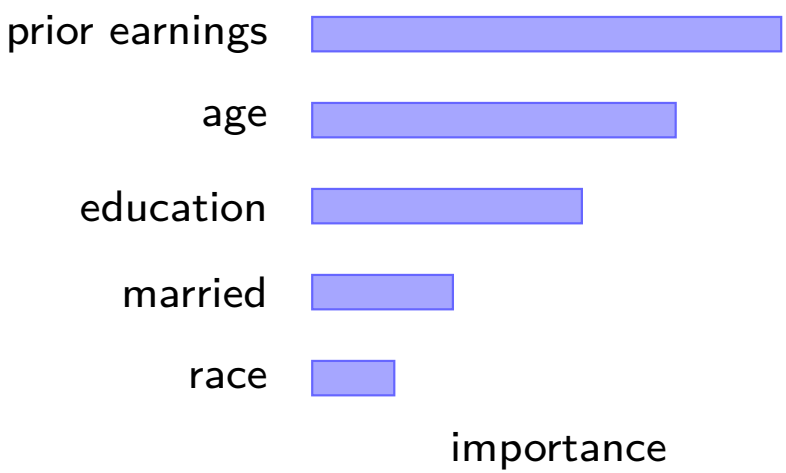
- Example interpretation:
  - $\hat{\beta}_1 < 0$: older workers tend to have smaller training effects.
  - $\hat{\beta}_3 < 0$: workers with higher previous earnings tend to benefit less.

Use: translates complex CATE estimates into familiar regression-style summaries.

After Causal Forest

Variable Importance

Question: Which covariates does the forest use most often to find treatment-effect heterogeneity?



- Counts how often variables are used in forest splits.
- High importance means the variable helps separate high-effect and low-effect groups.
- It can capture nonlinearities and interactions.
- But it does **not** tell us the direction of the relationship.

Use: identifies which variables are useful for discovering heterogeneity.

R Example

LaLonde Job Training Data

R Implementation

- Classic job training dataset from LaLonde (1986).
- Treatment D_i: participation in the training program.
- Outcome Y_i: earnings in 1978.
- Covariates X_i: age, education, race, marital status, no degree, earnings in 1974 and 1975.
- Main question: which workers gain more from training?

Data and Program

R Implementation

- See `causal_forest.R`
- Data: `lalonde` (built into the `Matching` package; no separate file needed)
- Install required packages:
  - `grf` — causal forest estimation (`causal_forest`, `variable_importance`, `best_linear_projection`)
  - `Matching` — includes the LaLonde dataset
  - `ggplot2` — CATE distribution and variable importance plots
  - `dplyr` — subgroup summaries
- Key steps: prepare (Y, D, X) $\rightarrow$ estimate forest $\rightarrow$ ATE/ATT $\rightarrow$ CATE $\rightarrow$ variable importance $\rightarrow$ export

Step 1: Package Installation and Data Loading

- Install and load required R packages:

```r
install.packages(c("grf", "Matching", "ggplot2"))

library(grf)
library(Matching)
library(ggplot2)
```

- Load the LaLonde data:

```r
data(lalonde)
summary(lalonde)
```

- `grf`: generalized random forests.
- `Matching`: includes the LaLonde dataset.
- `ggplot2`: visualization.

Step 2: Data Preparation

- Define outcome Y_i, treatment D_i, and covariates X_i:

```
Y <- lalonde$re78
D <- lalonde$treat
X <- lalonde[, c("age", "educ", "black", "hisp",
                 "married", "nodegree", "re74", "re75")]
X <- as.matrix(X)
```

- `Y`: observed outcome Y_i.
- `D`: treatment indicator D_i.
- `X`: pre-treatment covariates.
- All variables in `X` should be measured before treatment.

Step 3: Causal Forest Estimation

- Estimate the causal forest:

```r
set.seed(2026)

cf <- causal_forest(
  X = X,
  Y = Y,
  W = D,
  num.trees = 2000
)
```

- `causal_forest()` estimates heterogeneous treatment effects.
- `W = D`: `grf` names the treatment argument `W`, but our course notation is D_i.
- `num.trees = 2000`: more trees usually produce more stable estimates.
- The package handles honesty and forest averaging internally.

Step 4: Estimate ATE and ATT

- Report the average treatment effect and the effect on the treated:

```
ate <- average_treatment_effect(cf, target.sample = "all")
att <- average_treatment_effect(cf, target.sample = "treated")

ate
att
```

- Returns the estimated average treatment effect and standard error.
- `target.sample = "all"` estimates α_{ATE}.
- `target.sample = "treated"` estimates α_{ATT}.
- These connect the new method back to the earlier course estimands.

What ATE and ATT Mean Here

- The forest first estimates individual CATEs $\hat{\tau}(x_i)$.
- Then it averages them over different target samples:

$$\hat{\alpha}_{\text{ATE}} \approx \frac{1}{N}\sum_{i=1}^{N}\hat{\tau}(x_i), \qquad \hat{\alpha}_{\text{ATT}} \approx \frac{1}{N_1}\sum_{i:D_i=1}\hat{\tau}(x_i).$$

- In practice, `grf` uses forest weights and influence-function adjustments, so the package estimate is the preferred reported number.

Step 5: Get CATE Estimates

- Predict the CATE for each observation:

```
tau_hat <- predict(cf)$predictions

summary(tau_hat)
```

- `tau_hat` contains one estimated CATE for each observation.
- Interpretation: $\hat{\tau}(x_i)$ is the estimated training effect for workers similar to i.
- Use the summary to see whether effects look similar or very dispersed.

Step 6: Plot CATE Distribution

- Plot the distribution of estimated CATEs:

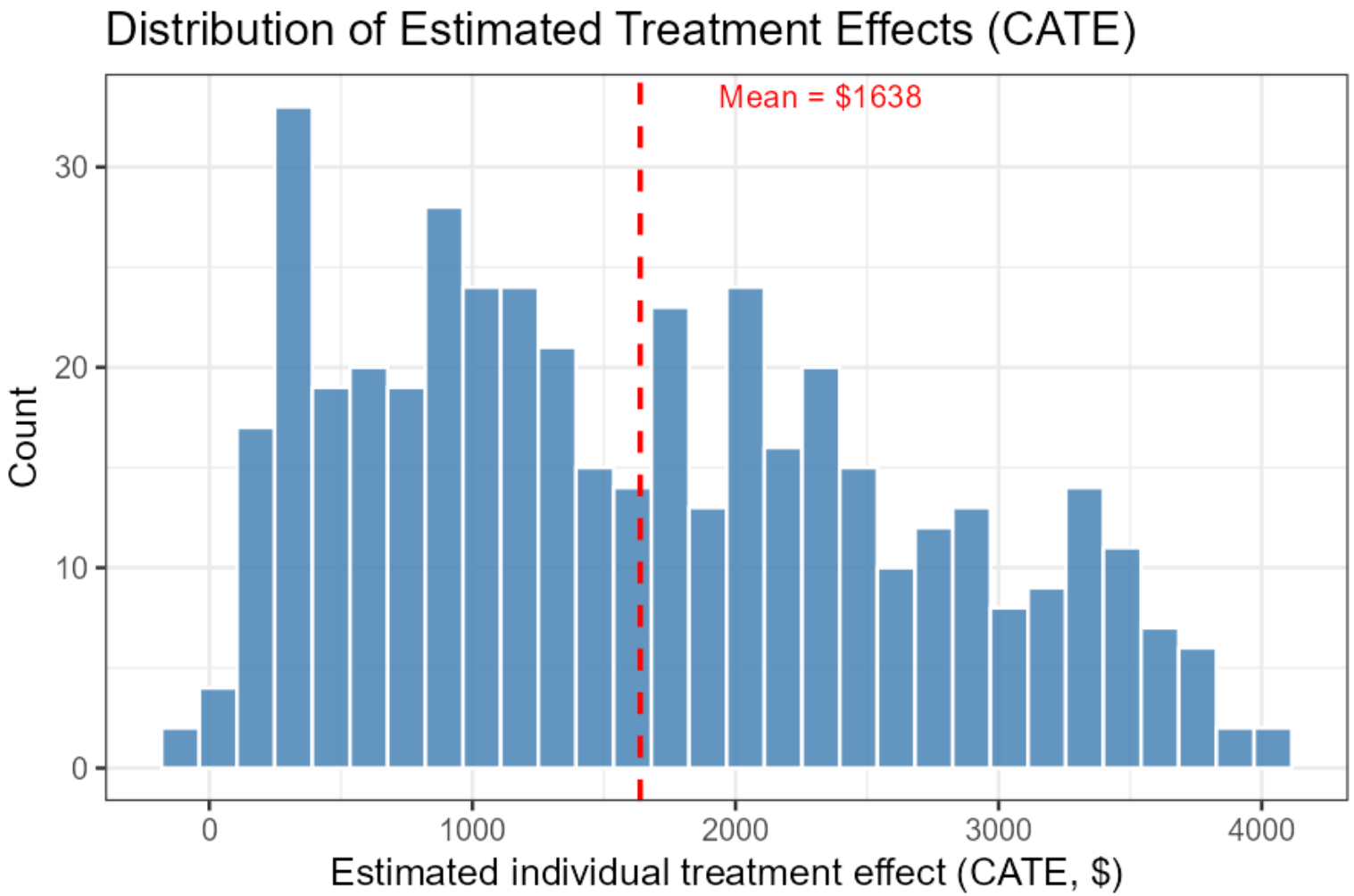
```r
cate_df <- data.frame(tau_hat = tau_hat)

ggplot(cate_df, aes(x = tau_hat)) +
  geom_histogram(bins = 30, fill = "steelblue",
                 color = "white") +
  geom_vline(xintercept = mean(tau_hat),
             color = "red", linetype = "dashed") +
  labs(x = "Estimated treatment effect",
       y = "Count")
```

- The red line is the average of the estimated CATEs.
- The spread shows the amount of estimated heterogeneity.

Step 6: CATE Distribution

LaLonde Job Training Data



- The red dashed line marks the mean CATE (close to the ATE estimate).
- The spread shows substantial heterogeneity: some workers benefit far more than others.

Step 7: Summarize Heterogeneity

- Use BLP and variable importance:

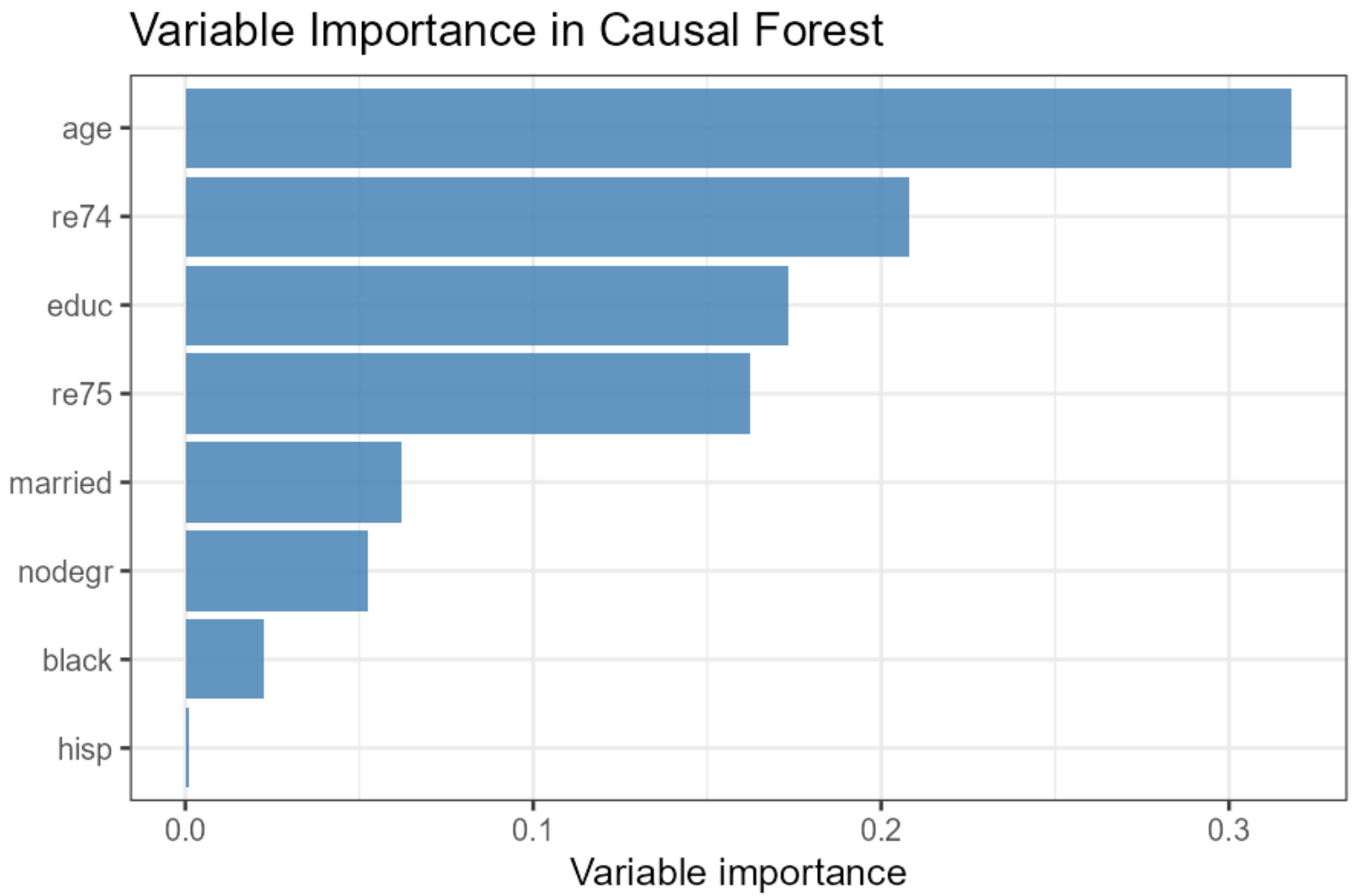
```r
# Best linear projection
blp <- best_linear_projection(cf, X)
blp

# Variable importance
vi <- variable_importance(cf)
names(vi) <- colnames(X)
sort(vi, decreasing = TRUE)
```

- BLP helps interpret which covariates are associated with larger effects.
- Variable importance helps identify which covariates are useful for splitting.
- Read them as summaries of heterogeneity, not as a replacement for the research design.

Step 7: Variable Importance

LaLonde Job Training Data



- **Age** and **pre-training earnings** (re74, re75) drive most of the splits.
- Workers who were younger or had lower pre-training earnings tend to benefit more from training.

Step 8: Export for Stata Post-Analysis

- Export CATE estimates for familiar post-analysis:

```r
results <- data.frame(
  id       = 1:nrow(X),
  tau_hat  = tau_hat,
  age      = lalonde$age,
  educ     = lalonde$educ,
  D        = lalonde$treat
)

write.csv(results, "cate_results.csv",
          row.names = FALSE)
```

- Estimate the forest in R.
- Export CATE estimates for familiar summary tables and plots elsewhere.

Recommended Resources

- Wager and Athey (2018), "Estimation and Inference of Heterogeneous Treatment Effects using Random Forests." *Journal of the American Statistical Association*, 113(523), 1228–1242.
- Athey, Tibshirani, and Wager (2019), "Generalized Random Forests." *The Annals of Statistics*, 47(2), 1148–1178.
- `grf` documentation:

  https://grf-labs.github.io/grf/

- Athey and Imbens (2019), "Machine Learning Methods That Economists Should Know About." *Annual Review of Economics*, 11, 685–725.